HANGMAN GAME

Xingrong Zong
AAA
02.04.2020

Contents

1. Revision History

Date	Version	Description	Author
02.04.2020	1	Project Plan	Xingrong Zong
	2	Test Plan	Xingrong Zong
	3	Project	Xingrong Zong
	4	Project	Xingrong Zong

2. General Information

Project Summary	
Project Name	Project ID
Hangman Game	20200202
Project Manager	Main Client
Xingrong Zong	Player
Key Stakeholders	
Project Manager Programmer / Game Designer	Player
Executive Summary	
The project Hangman Game is to create a program selects a word and the player is going to guess the word by guessing letter after letter. For every wrong guess, the game is building a part of a man getting hanged. The number of wrongs that the player can have is ten. The player loses if there are no guesses left. The game also ends if the player identifies all the words, and the player will be considered a winner and can choose if he/she wants one more round.	

3. Vision

The project Hangman Game is to create a word-guessing game. The game is written in Java.

For this text-based version, the player should be greeted with a menu to select either start or quit the game. When beginning the game, a word from a predefined list of words should randomly be picked and the number of letters displayed with equally many underscore signs.

Build the "image" of the hanging man using the available characters on the keyboard. For every wrong guess, the game is building a part of a man getting hanged. The number of wrongs that the player can have is ten. If there are no guesses left, the player fails the task and can choose to start another round or quit the game. If the player guesses the word within ten tries, the player will be considered a winner with restart choice as well.

Reflection: To create this vision document, the programmer and the game designer have to understand the concept and rules of the hangman game. The project planning includes some additional functions for the player,

4. Project Plan

4.1 Introduction

The project Hangman Game is to create a program selects a word and the player is going to guess the word by guessing letter after letter. For every wrong guess, the game is building a part of a man getting hanged. The number of wrongs that the player can have is ten. The player loses if there are no guesses left. The game also ends if the player identifies all the words, and the player will be considered a winner with a score shows based on player's correct guesses.

The project starts with project planning about tasks, time log.

4.2 Justification

The application is assigned to students participating in 1DV600 course as a project that aims to provide practical work with formal software project processing.

4.3 Stakeholders

Project Manager: Carries out project planning and scheduling to verify whether the work meets the requirements. Keeps track on the process of the project.

Programmer / Game Designer: Implements the code and designs functions to meet the functionalities and project's requirements.

Tester: Tests the product's functions. Checks if it meets the requirements or project planning and if there are any defects to develop.

Project Customer / End User: Player, people who will use the final product.

4.4 Resources

1DV600 Lectures (Videos)

Software Engineering by Ian Sommerville (Textbook)

4.5 Hard- and Software Requirements

Specify what is used to develop and later run the software developed.

Game runner: Windows 10,

Java Version 13

IDE: IntelliJ 2020

JDK

JRE

4.6 Overall Project Schedule

First iteration final deadline: 21/08/2020

Second iteration final deadline: 21/08/2020

Third iteration final deadline: 21/08/2020

Forth iteration final deadline: 21/08/2020

4.7 Scope, Constraints and Assumptions

Scope: The game will be running as a java class in IDE IntelliJ 2020. The project is text-based. The game can only be played by one player. Player can continue to play the game as many rounds as they want. Word is randomly selected from a predefined vocabulary library. The player has ten chances to guess the word.

Constraints: Due to time-consuming, developer's lack of practical skill and limited knowledge about developing and database, some preplanned functions can not be developed for the project, such as user registration, multiplayer, time limit and the ability to delete letter.

Assumptions: The player has some degree of English vocabularies knowledge and understands the rules of the hangman game. The player can have a delightful experience.

5. Iterations

5.1 Iteration 1

Implement idea and some skeleton code for the project. Generate words for game

Estimated time: 4 h

Actual time: 7 h

5.2 Iteration 2

Add some features to the game. Using UML.

Estimated time: 5 h

Actual time:

5.3 Iteration 3

Testing. Add additional features to the game in this iteration if needed.

Estimated time: 2 h

Actual time:

5.4 Iteration 4

Reiterate the steps in iteration 1 – 3 for a set of new features to make them as a whole project.

Estimated time: 1 h

Actual time:

6. Risk Analysis

6.1 List of risks

List the identified risks and specify, as far as possible, the probability of them happening as well as the impact they would have on the project.

Risk	Probability	Impact	Strategy
Underestimated development time	High	Serious	Wisely use time log to estimate developer's skills and time use. Plan the project more correctly and update the former if necessary.
Lack of knowledge	High	Tolerable	Gathering knowledge for the project from books and internet.
Requirements change	Moderate	Serious	Update the project plan as soon as possible, make sure it get solved before next change
Staff Emergency / Illness	Low	Serious	Try to do the work to save extra time for emergency. Almost no avoidance or minimization.
Hardware unavailability	Low	Serious	Save the code and plan work every time, and backup them on several platforms such as google drive and gitlab..

Reflection: I realize it is very important to make a preplanned project plan is the first step to start a project. It will save a lot of time and the probability that the risk will arise is reduced which increase productivity and efficiency. Preparation of minimizing the impact of them if they do occur and preparing for the worst and have a strategy in place to deal with it

7. Time log

Task	Estimated Time	Actual Time	Time Difference
Create a Project Plan	1.5h	2.5h	1h
Update the Project Plan			
Create a Time Log Table	5min	20min	15min
Update the Time Log table			
Create a Use Cases Diagram			
Update the Use Cases Diagram			
Create the fully dressed User Cases			
Update the fully dressed User Cases			
Create a State Machine Diagram			
Update the State Machine Diagram			
Create a Class Diagram			
Update a Class Diagram			
Read materials	2h	3h	1h
Update Read materials			
Implement the code for the Hangman Game	2h	5h	3h
Update the code			